



INTERNATIONAL STANDARD ISO/IEC 9075-4:2003 TECHNICAL CORRIGENDUM 2

Published 2007-04-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Database languages — SQL — Part 4: Persistent Stored Modules (SQL/PSM)

TECHNICAL CORRIGENDUM 2

Technologies de l'information — Langages de base de données — SQL —

Partie 4: Modules stockés persistants (SQL/PSM)

RECTIFICATIF TECHNIQUE 2

Technical Corrigendum 2 to ISO/IEC 9075-4:2003 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 32, *Data management and interchange*. Corrigendum 2 cancels and replaces ISO/IEC 9075-4:2003/Cor.1:2005.

Statement of purpose for rationale

A statement indicating the rationale for each change to ISO/IEC 9075-4:2003 is included. This is to inform the users of ISO/IEC 9075-4:2003 why it was judged necessary to change the original wording. In many cases, the reason is editorial or to clarify the wording; in some cases, it is to correct an error or an omission in the original wording.

Notes on numbering

Where this Technical Corrigendum introduces new Syntax, Access, General and Conformance Rules, the new rules have been numbered as follows:

Rules inserted between, for example, Rules 7) and 8) are numbered 7.1), 7.2), etc. [or 7)a.1), 7)a.2), etc.]. Those inserted before Rule 1) are numbered 0.1), 0.2), etc.

Where this Technical Corrigendum introduces new subclauses, the new subclauses have been numbered as follows:

Subclauses inserted between, for example, 4.3.2 and 4.3.3 are numbered 4.3.2a, 4.3.2b, etc. Those inserted before, for example, 4.3.1 are numbered 4.3.0, 4.3.0a, etc.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 9075-4:2003/Cor 2:2007

Contents

Page

Foreword.....	1
4 Concepts.....	1
4.8 Cursors.....	1
4.8.1 General description of cursors.....	1
4.13 Dynamic SQL concepts.....	1
4.13.1 Dynamic SQL statements and descriptor areas	2
5 Lexical elements.....	2
5.2 Names and identifiers.....	2
6 Scalar expressions.....	2
6.1 <value specification> and <target specification>	2
9 Schema definition and manipulation.....	3
9.2a <table definition>.....	3
9.2b <column definition>.....	4
9.3 <default clause>.....	5
9.24 <check constraint definition>.....	5
9.25 <view definition>.....	6
10 Access control.....	7
10.3 <revoke statement>.....	7
11 SQL-client modules.....	9
11.2 <SQL procedure statement>	9
12 Data manipulation.....	10
12.3 <close statement>.....	10
13 Control statements.....	11
13.1 <compound statement>.....	11
13.2 <handler declaration>.....	11
13.4 <SQL variable declaration>.....	12
13.5 <assignment statement>.....	12
13.9 <leave statement>.....	13
13.10 <loop statement>.....	13

13.11	<while statement>.....	13
13.12	<repeat statement>	13
13.13	<for statement>.....	14
17	Information Schema.....	15
17.2	MODULE_PRIVILEGES view.....	15
17.5	PARAMETERS view.....	15
18	Definition Schema.....	16
18.4	MODULES base table.....	16
Annex D (informative) Incompatibilities with ISO/IEC 9075-4:1999.....		17

Information technology — Database languages — SQL —

Part 4:

Persistent Stored Modules (SQL/PSM)

TECHNICAL CORRIGENDUM 2

Foreword

1. *Rationale: Remove incorrect reference to obsolete part.*

In the 7th paragraph, delete the 5th bullet.

4 Concepts

4.8 Cursors

4.8.1 General description of cursors

1. *Rationale: Clarify the specifications concerning dynamic result sets.*

Replace the 1st paragraph with:

Insert this paragraph For every <declare cursor> in a <compound statement>, a cursor is effectively created each time the <compound statement> is executed and destroyed when that execution completes.

NOTE 0.1 — Destroying an open with-result cursor does not simultaneously destroy that cursor's result set.

4.13 Dynamic SQL concepts

1. *Rationale: Clarification of the method of identification of dynamic objects.*

Insert the following new Subclause:

4.13 Dynamic SQL concepts

This Subclause modifies Subclause 4.24, “Dynamic SQL concepts”, in ISO/IEC 9075-2.

4.13.1 Dynamic SQL statements and descriptor areas

Replace the 8th paragraph A cursor declared by either a <declare cursor> or a <dynamic declare cursor> has a <cursor name>. A <dynamic declare cursor> is immediately contained in a <module contents>. A <declare cursor> is immediately contained either in the <module contents> of an <SQL-client module definition> or the <local cursor declaration list> of a <compound statement>. The scope of a <cursor name> is the innermost <SQL-client module definition> or <compound statement> that contains it.

5 Lexical elements

5.2 Names and identifiers

1. *Rationale: Provide replacement Syntax Rule for <cursor name> in Part 2.*

Insert the following Syntax Rule:

- 6) **Replace Syntax Rule 7)** Let CN be a <cursor name>. At least one of the following shall be true:
 - a) CN is contained, without an intervening <SQL schema statement> in an <SQL-client module definition> whose <module contents> contain a <declare cursor> or <dynamic declare cursor> whose <cursor name> is CN.
 - b) CN is contained, without an intervening <SQL schema statement>, in a <compound statement> whose <local cursor declaration list> contains a <declare cursor> whose <cursor name> is CN.

6 Scalar expressions

6.1 <value specification> and <target specification>

1. *Rationale: Add SQL variable to what a <target specification> and <simple target specification> can specify.*

Insert the following General Rules:

- 1) **Replace GR 3)** A <target specification> specifies a target that is a host parameter, an output SQL parameter, a column of a new transition variable, an element of a target whose declared type is an array type, a parameter used in a dynamically prepared statement, a host variable, or an SQL variable, according to whether the <target specification> is a <host parameter specification>, an <SQL parameter reference>, a <column reference>, a <target array element specification>, a <dynamic parameter specification>, an <embedded variable specification>, or an <SQL variable reference>, respectively.

- 2) Replace GR 13) A <simple target specification> specifies a target that is a host parameter, an output SQL parameter, a column of a new transition variable, a host variable, or an SQL variable, according to whether the <simple target specification> is a <host parameter specification>, an <SQL parameter reference>, a <column reference>, an <embedded variable name>, or an <SQL variable reference>, respectively.

NOTE 2.1 — A <simple target specification> can never be assigned the null value.

9 Schema definition and manipulation

9.2a <table definition>

1. *Rationale: Prohibit <SQL variable reference>s in DDL.*

Insert the following new Subclause:

9.2a <table definition>

This Subclause modifies Subclause 11.3, “<table definition>”, in ISO/IEC 9075-2.

Function

Define a persistent base table, a created local temporary table, or a global temporary table.

Format

No additional Format items.

Syntax Rules

- 1) Replace SR 0.1) The <table content source> shall not contain a <host parameter specification>, an <SQL parameter reference>, a <dynamic parameter specification>, an <embedded variable specification>, or an <SQL variable reference>.

Access Rules

No additional Access Rules.

General Rules

No additional General Rules.

Conformance Rules

No additional Conformance Rules.

9.2b <column definition>

1. *Rationale: Prohibit <SQL variable reference>s in DDL.*

Insert the following new Subclause:

9.2b <column definition>

This Subclause modifies Subclause 11.4, “<column definition>”, in ISO/IEC 9075-2.

Function

Define a column of a base table.

Format

No additional Format items.

Syntax Rules

- 1) Replace SR 0.1) The <column definition> shall not contain a <host parameter specification>, an <SQL parameter reference>, a <dynamic parameter specification>, an <embedded variable specification>, or an <SQL variable reference>.

Access Rules

No additional Access Rules.

General Rules

No additional General Rules.

Conformance Rules

No additional Conformance Rules.

9.3 <default clause>

1. *Rationale: Correct override of General Rules.*

Replace the General Rules with:

General Rules

No additional General Rules.

9.24 <check constraint definition>

1. *Rationale: Add <SQL variable reference> to what a <check constraint definition> cannot contain.*

Insert the following Subclause:

9.24 <check constraint definition>

Function

Specify a condition for the SQL-data.

Format

No additional Format items.

Syntax Rules

- 1) Replace SR 1) The <search condition> shall not contain a <host parameter specification>, an <SQL parameter reference>, a <dynamic parameter specification>, an <embedded variable specification>, or an <SQL variable reference>.

Access Rules

No additional Access Rules.

General Rules

No additional General Rules.

Conformance Rules

No additional Conformance Rules.

9.25 <view definition>

1. *Rationale: Add <SQL variable reference> to what a <view definition> cannot contain.*

Insert the following Subclause:

9.25 <view definition>

Function

Specify a condition for the SQL-data.

Format

No additional Format items.

Syntax Rules

- 1) Replace SR 3) The <view definition> shall not contain a <host parameter specification>, an <SQL parameter reference>, a <dynamic parameter specification>, an <embedded variable specification>, or an <SQL variable reference>.

Access Rules

No additional Access Rules.

General Rules

No additional General Rules.

Conformance Rules

No additional Conformance Rules.

10 Access control

10.3 <revoke statement>

1. *Rationale: Adjust references to rules which have moved in Part 2.*

Delete Syntax Rules 1), 2), 3), 4), 5), and 6).

2. *Rationale: Adjust references to rules which have moved in Part 2.*

Insert the following General Rules:

- 0.1) Insert after GR 0.15) e) EXECUTE privilege on every SQL-server module that includes one or more SQL-invoked routines that are among the subject routines of a <routine invocation> that is generally contained in the <query expression> of *V*.
- 0.2) Insert after GR 0.17) e) EXECUTE privilege on every SQL-server modules that includes one or more SQL-invoked routines that are among the subject routines of a <routine invocation> that is generally contained in any <search condition> of *TC*.
- 0.3) Insert after GR 0.18) e) EXECUTE privilege on every SQL-server module that includes one or more SQL-invoked routines that are among the subject routines of a <routine invocation> that is generally contained in any <search condition> of *AX*.
- 0.4) Insert after GR 0.20) e) EXECUTE privilege on every SQL-server module that includes one or more SQL-invoked routines that are among the subject routines of a <routine invocation> that is generally contained in any <search condition> of *DC*.
- 0.5) Insert after GR 0.29) a) EXECUTE privilege on every SQL-server module that includes one or more SQL-invoked routines that are among the subject routines of a <routine invocation> that is contained in the <routine body> of *RD*.
- 0.6) Insert after GR 0.32) Let *SSM* be any SQL-server module descriptor of an SQL-server module included in *S1*. *SSM* is said to be *abandoned* if the revoke destruction action would result in *A1* no longer having in its applicable privileges any of the following:
 - a) EXECUTE privilege on every schema-level routine that is among the subject routines of a <routine invocation> that is contained in the <routine body> of any SQL-invoked routine included in *SSM*.
 - b) EXECUTE privilege on every SQL-server module that includes one or more SQL-invoked routines that are among the subject routines of a <routine invocation> that is contained in the <SQL routine body> of any SQL-invoked routine included in *SSM*.
 - c) SELECT privilege on at least one column of each table identified by a <table reference> contained in a <query expression> simply contained in a <cursor specification>, an <insert statement>, or a <merge statement> contained in the <routine body> of any SQL-invoked routine with a security characteristic of *DEFINER* included in *SSM*.
 - d) SELECT privilege on at least one column of each table identified by a <table reference> contained in a <table expression> or <select list> immediately contained in a <select statement>:

single row> contained in the <routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.

- e) SELECT privilege on at least one column of each table identified by a <table reference> contained in a <search condition> contained in a <delete statement: positioned>, an <update statement: searched>, or a <merge statement> contained in the <routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- f) SELECT privilege on at least one column of each table identified by a <table reference> contained in a <value expression> simply contained in an <update source> or an <assigned row> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- g) SELECT privilege on at least one column identified by a <column reference> contained in a <search condition> contained in a <delete statement: searched>, an <update statement: searched>, or a <merge statement> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- h) SELECT privilege on at least one column identified by a <column reference> contained in a <value expression> simply contained in an <update source> or an <assigned row> contained in the SQL routine body of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- i) INSERT privilege on every column

Case:

- i) Identified by a <column name> contained in the <insert column list> of an <insert statement> contained in the <routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- ii) Of the table identified by the <table name> immediately contained in an <insert statement> that does not contain an <insert column list> and that is contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- iii) Of the table identified by the <target table> immediately contained in an <merge statement> that contains a <merge specification> and that does not contain an <insert column list> and that is contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- j) UPDATE privilege on every column whose name is contained in an <object column> contained in either an <update statement: positioned>, an <update statement: searched>, or a <merge statement> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- k) DELETE privilege on every table whose name is contained in a <table name> immediately contained in either a <delete statement: positioned> or a <delete statement: searched> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.
- l) USAGE privilege on every domain, every collation, every character set, and every transliteration whose name is contained in the <routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in SSM.

- m) USAGE privilege on every user-defined type *UDT* such that there is a <data type> contained in the <routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in *SSM* that is usage-dependent on *UDT*.
- n) The table/method privilege on every table *T1* and every method *M* such that there is a <method reference> *MR* contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in *SSM* such that *T1* is in the scope of the <value expression primary> of *MR* and *M* is the subject routine of *MR*.
- o) SELECT privilege WITH HIERARCHY OPTION on at least one supertable of the scoped table of any <reference resolution> that is contained in any <query expression> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in *SSM*.
- p) SELECT privilege WITH HIERARCHY OPTION on at least one supertable of the scoped table of any <reference resolution> that is contained in any <table expression> or <select list> immediately contained in a <select statement: single row> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in *SSM*.
- q) SELECT privilege WITH HIERARCHY OPTION on at least one supertable of the scoped table of any <reference resolution> that is contained in any <search condition> contained in a <delete statement: searched>, an <update statement: searched>, or a <merge statement> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in *SSM*.
- r) SELECT privilege WITH HIERARCHY OPTION on at least one supertable of the scoped table of any <reference resolution> that is contained in any <value expression> simply contained in an <update source> or an <assigned row> contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in *SSM*.
- s) SELECT privilege WITH HIERARCHY OPTION on at least one supertable of every typed table identified by a <table reference> that simply contains an <only spec> and that is contained in the <SQL routine body> of any SQL-invoked routine with a security characteristic of DEFINER included in *SSM*.

11 SQL-client modules

11.2 <SQL procedure statement>

1 *Rationale: Tags need to be revised to take into account corrections to Subclause 13.5, "<SQL procedure statement>" in ISO/IEC 9075-2.*

Replace the tag on General Rule 2) with:

Insert before GR 11)

2. *Rationale: Tags need to be revised to take into account corrections to Subclause 13.5, “<SQL procedure statement>” in ISO/IEC 9075-2.*

Replace the tag on General Rule 3) with:

Insert before GR 11)

3. *Rationale: Tags need to be revised to take into account corrections to Subclause 13.5, “<SQL procedure statement>” in ISO/IEC 9075-2.*

Replace the tag on General Rule 4) with:

Insert before GR 13)

12 Data manipulation

12.3 <close statement>

1. *Rationale: Correction of misplaced rule.*

In the Syntax Rule replace *No additional Syntax Rules* with:

- 1) Replace SR1) Let *CN* be the <cursor name> in the <close statement>. *CN* shall be contained within the scope of one or more <cursor name>s that are equivalent to *CN*. If there is more than one such <cursor name>, then the one with the innermost scope is specified. Let *CR* be the cursor specified by *CN*.

2. *Rationale: Correction of misplaced rule.*

Replace the General Rules with:

General Rules

No additional General Rules.

13 Control statements

13.1 <compound statement>

1. *Rationale: Provide a <beginning label> in the outermost scope.*

Replace Syntax Rule 2) with:

- 2) If CS does not specify a <beginning label>, then an implementation-dependent <beginning label> that is not equivalent to any other <statement label> contained in the outermost containing <SQL control statement> is implicit.

2. *Rationale: Clarify the specifications concerning dynamic result sets.*

Replace General Rule 3) c) ii) 1) with:

- 3) ...
 - c) ...
 - ii) ...
 - 1) For every open cursor *CR* that is not a with-return cursor that is declared in the <local cursor declaration list> of CS, the General Rules of Subclause 14.30, “Effect of closing a cursor”, are applied with *CR* as *CURSOR*.

3. *Rationale: Clarify the specifications concerning dynamic result sets.*

Replace General Rules 4) and 5) with:

- 4) For every open cursor *CR* that is not a with-return cursor that is declared in the <local cursor declaration list> of CS, the General Rules of Subclause 14.30, “Effect of closing a cursor”, are applied with *CR* as *CURSOR*.

NOTE 16 — “with-return cursor” is defined in Subclause 3.1.6, “Definitions provided in Part 2”, in ISO/IEC 9075-2.

- 5) The SQL variables, cursors, and handlers specified in <local declaration list>, the <local cursor declaration list>, and the <local handler declaration list> of CS are destroyed.

13.2 <handler declaration>

1. *Rationale: Clarify the specifications concerning dynamic result sets.*

Replace General Rule 3) b) iii) with:

- 3) ...

- b) ...
- iii) For every open cursor *CR* declared in *CS* that is not a with-return cursor, the General Rules of Subclause 14.30, “Effect of closing a cursor”, are applied with *CR* as *CURSOR*.

2. *Rationale: Clarify the specifications concerning dynamic result sets.*

Replace General Rule 3) c) iii) with:

- 3) ...
- c) ...
- iii) For every open cursor *CR* declared in *CS* that is not a with-return cursor, the General Rules of Subclause 14.30, “Effect of closing a cursor”, are applied with *CR* as *CURSOR*.

13.4 <SQL variable declaration>

1. *Rationale: Correct incorrect use of a “value” in a syntactic construct.*

Replace General Rule 1) with:

- 1) If <SQL variable declaration> contains <default clause> *DC*, then let *DV* be the <default option> contained in *DC*. Otherwise let *DV* be <null specification>. Let *SV* be the variable defined by the <SQL variable declaration>. The following SQL-statement is effectively executed:

SET *SV* = *DV*

13.5 <assignment statement>

1. *Rationale: Reference <mutated target specification> rather than <mutated target>.*

Replace Syntax Rule 5) with:

- 5) If <assignment target> immediately contains a <mutator reference>, then let *TS* be the <mutated target specification>, let *FN* be the <method name>, and let *AS* be the <assignment source>. The <assignment statement> is equivalent to:

SET *TS* = *TS.FN* (*AS*)

NOTE 19 — The preceding rule is applied recursively until the <assignment target> no longer contains a <mutator reference>.